Video Surveillance Framework Based on Real-Time Face Mask Detection and Recognition

Ehsan Nasiri
Computer Science
University of Arkansas at Little Rock
Little Rock,AR USA
Exnasiri@ualr.edu

Mariofanna Milanova
Computer Science
University of Arkansas at Little Rock
Little Rock,AR USA
Mgmilanova@ualr.edu

Ardalan Nasiri
Electrical Engineer
University of Arkansas at Fayetteville
Fayetteville,AR USA
Manasiri@uark.edu

2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA) | 978-1-6654-3603-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/INISTA52262.2021.9548475

***Abstract*— In this paper we proposed a real-time face mask detection and recognition for CCTV surveillance camera videos. The proposed work consists of six steps: video acquisition and keyframes selection, data augmentation, facial parts segmentation, pixel-based feature extraction, Bag of Visual Words (BoVW) generation, face mask detection, and face recognition. In the first step, a set of keyframes are selected using histogram of gradient (HoG) algorithm. Secondly, data augmentation is involved with three steps as color normalization, illumination correction (CLAHE), and poses normalization (Angular Affine Transformation). In third step, facial parts are segmented using clustering approach i.e. Expectation Maximization with Gaussian Mixture Model (EM-GMM), in which facial regions are segmented into Eyes, Nose, Mouth, Chin, and Forehead. Then, Pixel-based Feature Extraction is performed using Yolo Nano approach, which performance is higher and lightweight model than the Yolo Tiny V2 and Yolo Tiny V3, and extracted features are constructed into Codebook by Hassanat Similarity with K-Nearest neighbor (H-M with KNN) algorithm. For mask detection, L2 distance function is used. The final step is face recognition which is implemented by a Kernel-based Extreme Learning Machine with Slime Mould Optimization (SMO). Experiments conducted using Python IDLE 3.8 for the proposed Yolo Nano model and also previous works as GMM with Deep learning (GMM+DL), Convolutional Neural Network (CNN) with VGGF, Yolo Tiny V2, and Yolo Tiny V3 in terms of various performance metrics.**

Keywords*— *COVID-19 pandemic, Face Mask Detection, Face Recognition, Yolo Nano, Surveillance Video, KNN, and ArcFace.

I. Introduction

In recent years, face recognition is a widely researched topic that gives several applications. Face recognition from disguised/occluded / any other partially covered faces is a little difficult. Today, everyone wears a mask due to the spread of COVID-19 pandemic [1], [2], [3]. In such a problematic situation, one's covered face detection requires large efforts. But, we need to recognize the persons for timely taking actions [4], [5]. This research field is often known as "Masked Face Recognition". The current state-of-the-art works in this field are designed by deep learning approaches. The crucial attributes that must be a matter for masked face detection can include the following:

- Faces Type, need to know Ellipse or Circle
- Eyes Location, Mark Eye Centers
- Face Orientation, including Left, Left Front, Right, and Right Front
- Occlusion Degree, need to Define Four Regions (Eye, Forehead, and Eyebrows) [6], [7], [8].

However, Face Mask Detection Challenges, Rotated Faces, Foggy Environment, Un masked Faces, Crowd Area, Fully Covered Faces and Blurred Faces illustrates the challenges of classifying face-masked persons under COVID-19 period [9], [10]. To recognize a masked face, we must identify whether the human is wearing a mask or not [11], [12]. Recognizing masked faces and facial images taken from various sources such as video cameras, smartphones, CCTV surveillance cameras. From these sources, datasets are generated [13], [14], [15]. In video, two levels of features play a significant role include high level (motion) and low level (color, texture, shape, and demographic) [16], [17]. Demographic features mainly represent Race, Age, and Gender. In real-time surveillance cameras, keyframe extraction and selection is the most important operation. However, keyframes are varied in terms of contrast, brightness, color intensity, and resolution. Among the several facial regions, periocular region is one of the significant parts since it's uncovered by the medical masks. Hence, it is expected to have poor image quality. Besides other regions such as Nose region, Mouth Region, and Chin Region are important [18], [19], [20].

In order to address this issue, data augmentation-like preprocessing step is adopted. The core problem in masked face recognition is caused by the desired attributes of the publicly available real-world datasets which are listed as follows,

- Insufficient face images to recognize properly
- Most faces have uniform features. In such cases, it's difficult
- Variations in pixel intensity, illumination, occlusion in the captured facial images
- Segmentation of unmasked regions is probably extracted by clustering algorithms
- Deep learning by less complexity works is required, which reduces the training time.

II. Related Work

Authors in [21] propose classic machine learning and deep learning models for masked face recognition. The presented contains 2 elements as the main i.e., feature extraction and classification by means of feature extraction and classification, respectively. For feature extraction, ResNet50 is used whereas SVM, decision tree, and ensemble learning are used for classification. For testing, three kinds of datasets are considered i.e., real-world masked face dataset, simulated masked face dataset (SMFD), and labeled faces in the wild (LFW). This paper does not investigate the parameter tuning and kernel selection of SVM, which degrade the recognition accuracy in face mask detection. In [22], Gaussian mixture model (GMM) is presented for human face recognition. Features are extracted and analyzed by computing the similarity between the face samples. This step finds well the human face wear by mask and or not. The overall work is implemented using OpenCV where used *dlib library* for

978-1-6654-3603-8/21/$31.00 ©2021 IEEE

implementing deep learning algorithms. When compared with other traditional methods of face recognition, the GMM based model shows good performance that recognizes abnormal faces such as masks, respirators, and sunglasses. GMM based feature extraction often needs of dynamic set of threshold values and Haar classifier does not suitable for low-resolution images. A convolutional neural network (CNN) [23] is proposed which uses periocular region of the face. In the first step, facial region is detected, and facial landmarks are pointed out. Then eye center is computed by using facial landmarks and thus loose and tight periocular regions are detected. Each periocular region, individual CNN is applied and score values are computed. Finally, a score level fusion is calculated. A custom-made Dongguk Periocular Database (DP-DB1) and Choke Point Database are used for experimentation. Even when an occlusion is presented, face recognition shows by this paper has increased. However, closed eyes, occlusions, and posture changes are not predicted by the proposed CNN, since max pool layer does not consider these changes.

The periocular region is a significant part of masked face recognition. In [24], a deep learning model is presented for accurate face recognition in which an attention module is incorporated for semantic regions (eye and eyebrow) detection. In particular, AttNet and FCN – Peri is used which have the same configuration. In between convolution layer, attention layer is constructed. A customized verification-based loss function is applied which provides high discriminating power than other loss functions such as triplet / contrastive. Each input image generates ROI which must be accurate. Otherwise, accuracy reached too very low.

In [25], face recognition is implemented for video frames. Face recognition between normal and abnormal faces in existing systems uses a single image. With the use of single image, faces are not identified since its pose orientation and illumination problems. To address this serious issue, face detection is performed by video frames. In each frame, a libface detection model is used to detect faces. For training and testing, GMM algorithm is applied. In GMM, expectation-maximization (EM) is used, in which probabilistic values are computed. Three different sets of images are used for performance analysis.

III. Problem Statement

Masked face detection and recognition is an emerging medical field. Very little research is conducted in this field, but it's been popular recently due to the existence of COVID-19 pandemic. Recognition accuracy is a crucial element that still suffers from the absence of primary tasks. Certain most important challenges are given below.

- Lack of Data Augmentation - Data augmentation is required before masked face recognition. For example, pose normalization is one of the significant steps in data augmentation since head poses may variant for humans. Pose normalization results front face which covers most face regions. Similarly, it requires other data augmentation tasks such as color normalization and illumination correction. Existing works failed to perform these tasks.
- Skin Texture Analysis Nonexistence – Human skin textures must be unique to one another. It has unique lines, pores, and spots appearance. In masked faces, we can't analyze skin texture, but it's possible to extract partial face images when it's covered.
- Absence of Accurate Loss Function in Object Detection Model – In literature, Fast Region-based CNN, Yolo V2, YoloV3, and tiny versions are proposed object detection. The size of architecture is very large, which is heavyweight. It is a widely used deep learning algorithm for masked face recognition, but it does not suit real-time face recognition. In particular, SoftMax Loss is higher in CNN, and also it suffered from inter and intra class separation problems.

The specific problems that are taken into account in this paper are as follows: Authors in [26] have presented face recognition in partially covered faces. Hybrid CNN and VGGF algorithms are presented for feature extraction and matching. For training and validation, masked faces, occluded faces, zoomed-out faces, and disguised faces are focused. The most several problems in partially covered face recognition are as follows:

- If face image is rotated and oriented, CNN-based classifiers have obtained poor performance. That is to say, Position and Orientation of a given input image are ignored in the max-pooling layers of CNN algorithm. This information is highly needed for feature extraction. Hence, it has less accuracy in classification.
- SVM is slower in processing which tends to cause higher processing time, especially has higher training time hence this is not applicable for real-time face recognition systems. Furthermore, parameter tuning and kernel selection must be optimum.
- SoftMax loss does not have discriminant power for class separation (Inter and Intra Classes) which does not suit deep face recognition.
- Head poses are different in dataset, which does not sufficient to masked face recognition. And, skin texture was not analyzed which reduces the recognition accuracy.

Human skin is varied [27] for masked and unmasked images and also regions. Hence, color conversion is implemented to predict the color values of the input image.

Most of the works in face mask detection have used Yolo, CNN algorithms [28]. In Yolo, wide variety of algorithms that are Yolo Tiny V2, V3 are used. Though, it has several issues as,

- Yolo's previous versions have several limitations such as (1). The limitation of grid cells since it is hard to detect, (2). It has a lower ability for bounding box detection, which does not optimally change according to human pose.
- This work is greatly affected by artificial factors such as pixel intensity, illumination, and variance in head poses.

In masked faces, periocular region has played a significant role, which gives the accurate recognition result [29].

- Texture features only do not increase accuracy in periocular region. Dual-stream CNN increases complexity for extracting features and fusion and also long training time is needed in this combined work.

- Further, local binary-coded pattern does not work well when lightening conditions and is also less robust under disguised and partially covered faces. It produces long histograms which slow down the recognition speed, particularly for large training databases.

IV. Methodology

A. Architecture Overview

Our proposed work overthrow problem arises in existing masked face recognition. Our framework is composed of several sequential processes that are Key Frame Selection, Data Augmentation, Face Regions Segmentation, and Horizontal Slicing, Multi-Feature Extraction, Bag of Visual Words (BoVW) and Visual Words Construction, Masked Face Detection and Classification. Fig.1 illustrates the overall proposed model.

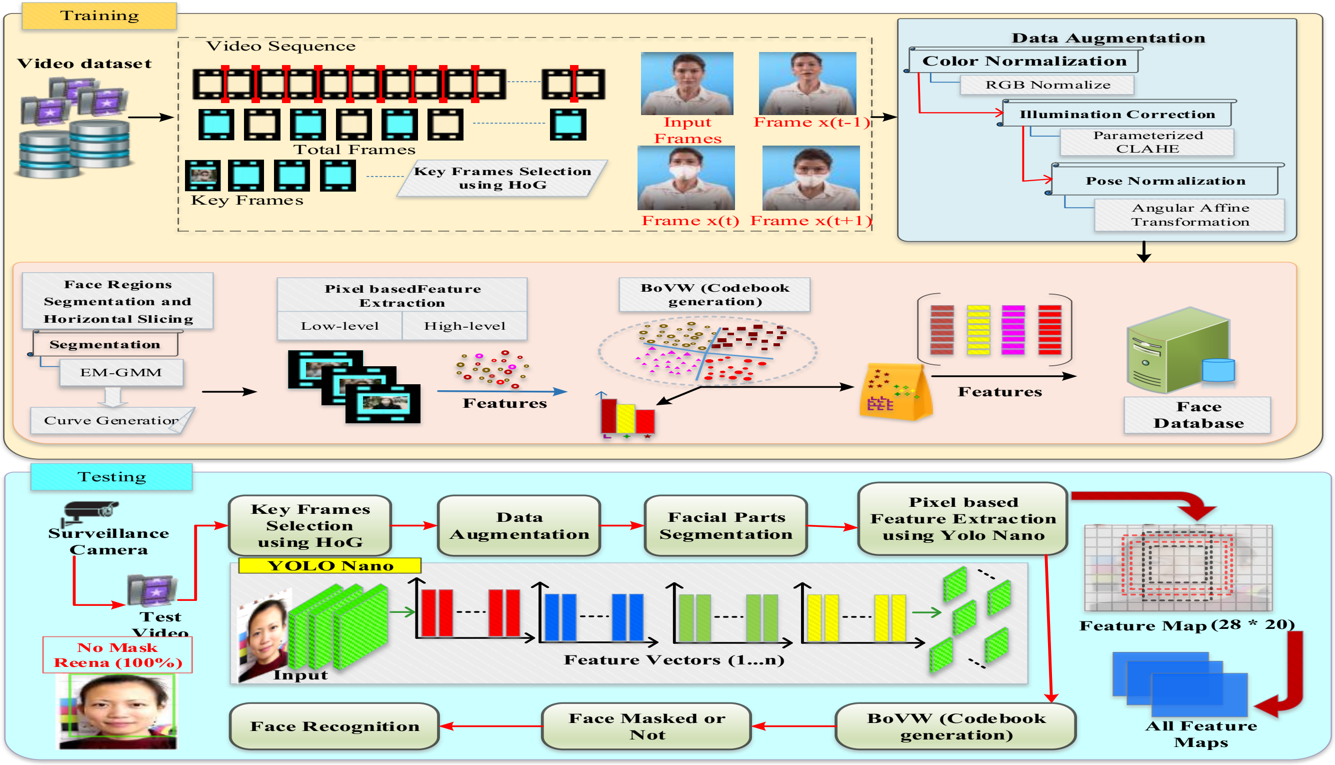

Fig.1. System Architecture

A CCTV surveillance video is loaded into the system for processing masked face detection and recognition. A detailed description for the proposed work is given as follows,

B. Key Frame Selection

Human keyframes are selected using a histogram of gradients (HoG). In this HoG, a Gaussian filter is used to find the difference between two frames and predict the human faces frames, which are called keyframes.

C. Data Augmentation

Our data augmentation step contains three processes that are color normalization, illumination correction, and pose normalization.

a) Color normalization

In color normalization step, pixel intensity distribution is performed which results normalized for R, G, and B channels. RGB is a color model that consists of three color components as RED, GREEN, and BLUE. It is represented as additive primitives and the color combination function is derived by follows,

$$\varsigma_p = R_p i + G_p j + B_p k \quad (1)$$

From the above equation, RGB color values are combined into a single color value and this combined value plays a vital role in feature extraction for accurate recognition results.

b) Illumination Correction

We adopt Parameterized Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm for illumination normalization that removes illumination in given input image. The proposed PCLAHE model uses the luminance and contrast parameters adaptively for each frame. The Gamma correction method is used to establish the dark areas of the given image. It improves the whole luminance of the given image block. A dynamic range of gamma correction for each block is represented by follows,

$$\beta = \frac{p}{d_r}\left(1+\tau\frac{g_{max}}{R}+\frac{\alpha}{100}\left(\frac{\sigma}{A_v+c}\right)\right) \quad (2)$$

Here, p demonstrates the number of the pixels in each block, d_r means dynamic range of this block. τ and α represent the stable parameters that are used to control the weight of dynamic range and entropies. σ is mentioned as the standard deviation of the block, A_v point out mean value, and c is the small value to avoid division by 0. R is dynamic value of luminance for a whole image. g_{max} means the maximum pixel value of the image. The gamma corrections are introduced to adjust the contrast value based on the current luminance value.

c) Pose Normalization

Image databases contain different poses in image, hence without performing proper pose normalization tends to have low accuracy in classification. Pose normalization was carried out using Angular Affine Transformation algorithm. We initially estimate pose angle of given image using Angle (Yaw, Pitch, and Roll). Then, the estimated angle is provided to the Affine Transformation to get a frontal view of the given image. Image cropping is performed after completion of pose normalization in order to maintain the same size for all input images.

D. Face Regions Segmentation and Horizontal Slicing

Unmasked face regions segmentation and horizontal striping are introduced to segment facial images. Here, the clustering-based segmentation is performed using the Expectation-Maximization based Gaussian Mixture Model (EM-GMM) algorithm. With EM-GMM, curve is generated. Each time a new curve is generated. It overwhelms the problems of fuzzy c means algorithm while segmenting facial parts. This way of segmentation tends to easier the process of classification. In EM-based GMM, similar pixel values are gathered and then we integrate the two clustering approaches such as EM and GMM. In GMM, *gaussian mixture* represents the linear superposition of Gaussians which follow,

$$p(x) = \sum_{k=1}^{K} \pi_k M(x|\mu_{k,}\Sigma_k) \qquad (3)$$

Where K is the total number of Gaussians, k is the mixing coefficient and weightage for each Gaussian distribution. In this work, EM purpose is to perform the iterative optimization which is performed locally. In the clustering technique, two processes are considered as,

- Expectation: For the input parameters set, we compute the *latent variable* expected values
- Maximization: According to the latent variables, the values of parameters are updated.

In GMM algorithm, likelihood function maximization is a significant part in which *mean and covariance* components are measured. EM-GMM based clustering procedure as follows,

(i) Initialize the mean, covariance, and mixing components μ_j, Σ_j, and π_j respectively, and then compute the initial loglikelihood value

(ii) Implement the estimation step in which we calculate the tasks using the current metrics as follows

$$\gamma_j(x) = \frac{\pi_k M(x|\mu_k,\Sigma_k)}{\sum_{k=1}^{K} \pi_j M(x|\mu_{j,}\Sigma_j)} \qquad (4)$$

Then perform the maximization step in which recomputed the current parameters by the following.

$$\mu_j = \frac{\sum_{n=1}^{N} \gamma_j(x_n) x_n}{\sum_{n=1}^{N} \gamma_j(x_n)} \qquad (5)$$

$$\Sigma_j = \frac{\sum_{n=1}^{N} \gamma_j(x_n)(x_n-\mu_j)(x_n-\mu_j)^T}{\sum_{n=1}^{N} \gamma_j(x_n)} \qquad (6)$$

$$\pi_j = \frac{1}{n} \sum_{n=1}^{N} \gamma_j(x_n) \qquad (7)$$

(iii) Next, we compute the Log-Likelihood

$$\ln p(X|\mu,\Sigma,\pi) = \sum_{n=1}^{N} ln\{\sum_{k=1}^{K} \pi_k M(x_n|\mu_{k,}\Sigma_k)\} \qquad (8)$$

(iv) Segments of face parts are implemented using similar set of clusters. Fig 2 demonstrates the performance clustering for EM-based GMM clustering.

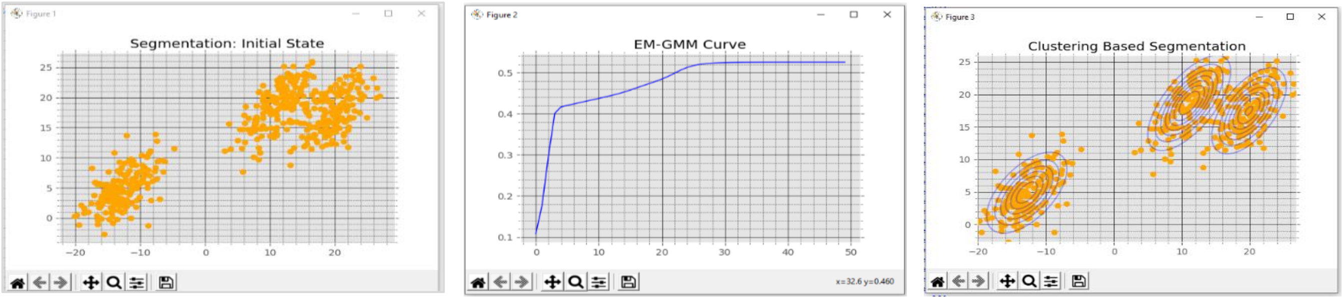

Fig.2.EM-GMM Algorithm

E. Yolo Nano based Multi-Feature Extraction & Masked Face Detection

Our feature extraction process extracts three types of features from the image to improve performance of classification and hence it's called multi-feature extraction. Feature extraction phase uses supervised learning and data are labeled data. Here, features are extracted from four regions are Periocular (forehead, eyes (left & right), Nose, Mouth, and Chin**.** For this purpose, we propose Yolo Nano to extract features from the given segmented parts. The prime advantage of Yolo NANO over CNN, F-RCNN, Yolo V1, V2, V3, and Tiny is that its robustness in extracting the position and orientation information of the given image which is required to classify the masked image accurately and further it is a lightweight architecture requires very less amount of time for training. Loss computed by using ArcFace (Additive Angular Margin Loss).

High-level features

- Motion (Any Feature or Object Changes over a Time)
- Spatial (Position, Angle, Orientation)

Low-level features

- Color (Color Channel Values – HSV)
- Shape (Facial Parts Shape Values)
- Texture (Skin Surface and Appearance Type)

The most important extracted features listed below,

- Mean: It is represented as the pixel intensity distribution in the whole region.

$$\mu_i = \frac{1}{n} \sum_{j=1}^{n} P_{ij} \qquad (10)$$

- Variance: It is determined how each pixel can be varied from the center or neighboring pixels.

$$\sigma_i = \sqrt{\left(\frac{1}{n} \sum_{j=1}^{n} (P_{ij} - \mu_i)^2\right)} \qquad (9)$$

- Skewness **–** It represents the symmetry measure for the given face image and it defines when the pixel values occur at the regular interval.

$$\mathrm{SW_i} = \sqrt{\left(\frac{1}{\mathrm{n}}\sum_{\mathrm{j=1}}^{\mathrm{n}}\left(\mathrm{P_{ij}} - \mathrm{\mu_i}\right)^3\right)} \qquad (11)$$

- Standard Variance: This is the second-moment average value computed from the number of pixels.

$$\sigma = \sum_{\mathrm{i=0}}^{\mathrm{l-1}}(\mathrm{p_i} - \mathrm{\mu})^3 \times \mathrm{H(p_i)} \qquad (12)$$

Texture features are defined as the surface or appearance measurement for a given object and it predicts the intensity, edges, and direction of pixels. Some of the texture features are described in the following.

- *Energy:* It is defined as the sum of square values of the pixels, which is also known as Uniformity or angular second moment. In mathematically, it is expressed as follows

$$E_p = \sum_i \sum_j p^2(i,j) \qquad (13)$$

- *Correlation:* This measures the color values dependency between the neighboring pixels.

$$C_p = \sum_i \sum_j p(i,j)logp(i,j) \qquad (14)$$

- *Contrast:* The intensity contrast is measured between the current and neighboring pixel values. It is computed by follows,

$$CT_p = \sum_i \sum_j (i-j)^2 p(i,j) \qquad (15)$$

- *Homogeneity:* It is inversely proportional to the contrast value and it represents the equivalent distribution of pixels over the region.

$$H_p = \sum_i \sum_j \frac{p(i,j)}{1+|i-j|} \qquad (16)$$

Yolo Nano architecture is a classic family of Yolo and it provides the expected outcomes by single-stage process since the balance between the inference time and accuracy. Yolo Nano model is inspired by the Yolo v3 network and it tries to achieve real-time face recognition that similar to CCTV surveillance videos processing. From Yolo V3, it is improved in two ways as (1). Slimmer Structure and (2). Instance Normalization.

a) Slimmer Structure of Yolo Nano

The macro structure of Yolo Nano is three components based such as Projection, Expansion, and Projection (PEP) module, Expansion Projection (EP) module, and Fully Connected Attention (FCA) module. Compared to other Yolo models, many PEP layers are eliminated to make the network lightweight and the processes are reduced by 50%.

b) Instance Normalization

In this step, contrast of the image is improved and content of low-level features are preserved such as textures and strokes. However, instance normalization is represented as follows,

$$Y_{ncij} = \frac{X_{ncij}-\mu_{nc}}{\sqrt{\sigma_{nc}^2+\in}} \qquad (17)$$

$$\mu_{nc} = \frac{1}{hw}\sum_{l=1}^{w}\sum_{m=1}^{h} X_{NCLM} \qquad (18)$$

$$\sigma_{nc}^2 = \frac{1}{hw}\sum_{l=1}^{w}\sum_{m=1}^{h}(X_{NCLM} - \mu_{NC})^2 \qquad (19)$$

Where X_{ncij} represents the $ncij$ is the layer feature map, i and j are the spatial dimensions, c is the channel feature, n represent the nth image in the batch. ∈ is the small integer number that is involved to eliminate more number of computations. μ_{nc} is the mean value of feature image in the i the image and σ_{nc}^2 is the variance value of feature image in the ith image.

BoVW Model: For extracted features, BoVW is implemented which constructs visual words dictionary and reduce feature space by clustering similar features using HS with KNN algorithm. KNN is a nearest neighbor prediction algorithm that can find the adjacent neighbor based on the high probability value. For visual words generation and code book generation, in this paper, we presented KNN algorithm. In KNN, the distance between one feature to another is computed. The traditional KNN uses Euclidean distance that produces more noise. And it does not produce low-distance precision. It must be trained properly to eliminate the noise and low precision issues. Therefore, in this paper, we selected the most adopted distance formula in KNN i.e. Hassanat distance. It provides better performance in code book generation.

Algorithm for HM-KNN

```
Input: Set of features
1. Begin
2. State the feature vector sets for training set $t_S$
3. For each frame $F_i$ feature set $F_S$ do
4. Visual words generation……
     (a). Initialize $K$  // $K =$ Small integer value
     (b). Compute the distance between $F_i$ and $t_S$
     (c). Choose $K$ in $t_S$ close to $F_i$
     (d). Assign the most similar class close to the distance to $F_i$
5. Compute the class label for all frames of features
6. End for
Output: Assign the class label for all frames in a video $V$
```

Algorithm description is as follows: the training set of videos is denoted as t_s and the class label for each input frame F_I is stored in database. Then, Hassanat similarity function is used to assign the exact class of the input frame. Based on that, the nearest visual words are constructed into a single group. The HM distance function is calculated as follows,

$$HD(X,Y) = \sum_{i=1}^{N} D(X_i, Y_i) \qquad (20)$$

Where $D(X,Y) =$

$$\begin{cases} 1 - \frac{1+Min(X_i,Y_i)}{Max(X_i,Y_i)}, & Min\ (X_i,Y_i) \geq 0 \\ 1 - \frac{1+Min(X_i,Y_i)+|Min(X_i,Y_i)|}{Max(X_i,Y_i)+|Min(X_i,Y_i)|} & Min\ (X_i,Y_i) < 0 \end{cases}$$

$D(X,Y)$ is bounded by 0 and 1.

In BoVW model, facial features f_i $(1,\ldots N)$ are constructed into a set of local keypoint descriptors as $f_i^p = \{p_{i,1}, p_{i,2}, p_{i,3}, \ldots, p_{1,m}\}$. Thus the BoVW model is defined by follows,

$$BoVW\colon R^d \to [1,N] \qquad (21)$$

$$P_{i,j} \to BoVW(p_{i,j}) \qquad (22)$$

Where, $p_{i,j} \in R^d$ is a mapping descriptor that was used to produce an integer index.

Finally, the masked face is detected (wears mask or not) by computing the distance using L2 distance function. We fetch the weights from masked values of the testing image and the trained image from HS-KNN. For similarity computation to classify the testing label L_2 distance is used,

$$D = \sqrt{\sum_{x=1}^{k}\left(VCt_{xi} - VCt_{xj}\right)^2} \qquad (23)$$

Where, $x = \{1,2,3 \ldots . k\}$, and VC_t - visual codebook of image.

F. Kernel ELM for Classification

The classification is significant process in the masked face detection of the given extracted features. In this, kernel-based extreme learning machine (ELM) with SMO algorithm is used to classify it. To optimize the performance of Kernel-based ELM, SMO algorithm is used. Based on the given input, proposed classification algorithm classifies the masked faces person.

The classification of the face and non-face region is carried out in the hidden node using the output weight W. $F_L(x) = \sum_{n=1}^{L} \alpha_n \mu_n(X)$ where α_n denotes the output weight of the nth hidden node. $J(x) = [j_n(X), .., j_L(X)]$ is the hidden layer output of ELM. Given N the video frame, the hidden layer output matrix J of ELM is given as,

$$J = \begin{bmatrix} j(X_1) \\ \vdots \\ j(X_N) \end{bmatrix} = \begin{bmatrix} G\ (p_1, q_1, X_1) & \cdots & G\ (p_L, q_L, X_L) \\ \vdots & \ddots & \vdots \\ G\ (p_1,\ q_1, X_N) & \cdots & G\ (p_L, q_L, X_N) \end{bmatrix} \quad (24)$$

Where TM is the training matrix: $TM = \begin{bmatrix} r_1 \\ \vdots \\ r_N \end{bmatrix}$ (25)

The objective of ELM is to minimize,

$$\|\alpha\|_{a}^{\gamma_1} + C\|\ J\alpha - TM\|_{b}^{\gamma_2} \quad (26)$$

Where $\gamma_1, \gamma_2 > 0, a, b = 0, \frac{1}{2}, 1,2, \dots, +\infty$.

V. Experimental Results

In this section, we describe the performance of the proposed Yolo Nano model with comparison to the GMM+DL [22], CNN with VGGF [26], Yolo Tiny V2, and Yolo Tiny V3. Further, the performance metrics are listed as follows.

a) Classification Accuracy

Classification accuracy is a significant metric that is most suitable for denoting the classified result. It is computed by four terms as True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) and accuracy is computed by follows.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (27)$$

The evaluation of accuracy for both face mask and non-face mask images is not relevant. When compare to non-masked faces, masked face detection is a critical task, and improving accuracy level for masked faces shows greater achievement of the presented model. In this study, we can expect higher classification accuracy owing to the Yolo Nano model, and also segmentation is performed in an accurate way.

b) Precision

In this metric, the performance of accurately classified face masks and non-masks are measured.

c) Recall

Recall is the proportion of positive cases that are determined accurately. In other words, it is the fraction of relevant images that are successfully determined. It is additionally referred to as true positive.

d) F-Measure

It is the measure of precision and recalls value combination and it is also known as F1-measure and F-score. In particular, it is the mean value of precision and recall.

e) ROC Curve

A Receiver Operating Curve (ROC) is a graph used for system organization and visualization. It is a distinct option used for recall and precision curves. ROC curves are normally used in medical diagnosis decision making and current years it is used for COVID-19 detection in more.

f) Computational Time

It is the sum of time taken to process the inputs for specific processes to reach the expected outcome. Previous methods consume more timing for training and testing, which results the sum of time taken to process the image is very high. We found that Yolo-Nano model produces highest accuracy, precision, recall, and f-measure in more consistent computation time. The number of layers in Yolo Nano is very less and sequential operations do not produce high processing and computational time

g) Confusion Matrix

It is the matrix to deal with the face mask detection and recognition evaluation that revolved with the ground truth results with the obtained results. The ROC curve reveals that the correlation between the TPR and FPR and differentiate the face and non-face classes in the dataset. In other words, confusion matrix C is a square matrix where $C(ij)$ denotes the number of faces that are known in the video i (True Label) and predicted to be in group j (Predicted Label).

A. Novelty & Significance Analysis

The novelty and significance of the proposed work are listed as follows:

- We presented Kernel-based ELM with an SMO algorithm, which improves recognition accuracy by greater than 4% of SVM since its, learning is better and also speed fast.
- Arcface based Yolo Nano is proposed which uses additive marginal loss, which performance is higher when performing deep face recognition
- Multiple features are extracted such as color tones, texture, and shape for recognition of masked faces.
- Illumination and non-uniform pixel intensity are the most important artifacts that remove completely in the data augmentation stage.

VI. Conclusion

In this paper, Yolo Nano objection model is used for face mask detection and recognition. For that, data augmentation, feature extraction, codebook generation, face mask detection, and recognition. We evaluated the performance of Yolo Nano model according to the accuracy, precision, recall, f-measure, and computational time on the CCTV surveillance video. In this paper, we explore the issues of disguised images such as illumination variation, noise, scale, pose in a single frame, and more patches variation. Based on the research gaps, algorithm comparison, and multi-views of performance are analyzed for the proposed and previous methods. The desired face detection model has higher results in terms of accuracy and precision in less computational time. This is very interesting part of the proposed work and also we think that due to algorithm selection, a process considered and more facial features of different views of human frontal faces in any angular degrees improve the performance of face mask detection and recognition.

To work in future extension of this paper, we can focus on the following research areas:

- We apply the proposed model for other applications such as facial expression recognition is concentrated to identify emotions for mask wear people
- We use more information fusion and transformation methods to further improve detection and recognition performance.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation under Award No. OIA-1946391(DART) and NSA grant NCAE -C Cyber Curriculum and Research 2020 Program, NCAEC-003-3030

REFERENCES

[1] Cheng, V. C. C., Wong, S.-C., Chuang, V. W. M., So, S. Y. C., Chen, J. H. K., Sridhar, S., … Yuen, K.-Y. (2020). The role of community-wide wearing of face mask for control of coronavirus disease 2019 (COVID-19) epidemic due to SARS-CoV-2. Journal of Infection.

[2] Cabani, A., Hammoudi, K., Benhabiles, H., & Melkemi, M. (2020). MaskedFace-Net – A dataset of correctly/incorrectly masked face images in the context of COVID-19. Smart Health, 100144.

[3] Razavi, M., Alikhani, H., Janfaza, V., Sadeghi, B., & Alikhani, E. (2021). An Automatic System to Monitor the Physical Distance and Face Mask Wearing of Construction Workers in COVID-19 Pandemic.

[4] Ejaz, M. S., & Islam, M. R. (2019). Masked Face Recognition Using Convolutional Neural Network. 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI).

[5] Meenpal, T., Balakrishnan, A., & Verma, A. (2019). Facial Mask Detection using Semantic Segmentation. 2019 4th International Conference on Computing, Communications and Security (ICCCS).

[6] Bhuiyan, M. R., Khushbu, S. A., & Islam, M. S. (2020). A Deep Learning-Based Assistive System to Classify COVID-19 Face Mask for Human Safety with YOLOv3. 2020 11th International Conference on Computing, Communication, and Networking Technologies (ICCCNT).

[7] Bu, W., Xiao, J., Zhou, C., Yang, M., & Peng, C. (2017). A cascade framework for masked face detection. 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation, and Mechatronics (RAM).

[8] Joshi, A.S., Joshi, S.S., Kanahasabai, G., Kapil, R., & Gupta, S. (2020). Deep Learning Framework to Detect Face Masks from Video Footage. *2020 12th International Conference on Computational Intelligence and Communication Networks (CICN)*, 435-440.

[9] Draughon, G., Sun, P., & Lynch, J. (2020). Implementation of a Computer Vision Framework for Tracking and Visualizing Face Mask Usage in Urban Environments. *2020 IEEE International Smart Cities Conference (ISC2)*, 1-8.

[10] Kose, N., & Dugelay, J.-L. (2014). Mask spoofing in face recognition and countermeasures. Image and Vision Computing, 32(10), 779–789.

[11] Qezavati, H., Majidi, B., & Manzuri, M.T. (2019). Partially Covered Face Detection in Presence of Headscarf for Surveillance Applications. 2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA), 195-199.

[12] Yuan, C., & Yang, Q. (2016). A Dynamic Face Recognition Deploy and Control System Based on Deep Learning. Journal of Residuals Science & Technology, 13.

[13] Engoor, S., Selvaraju, S., Christopher, H.S., Suryanarayanan, M.G., & Ranganathan, B. (2020). Effective Emotion Recognition from Partially Occluded Facial Images Using Deep Learning.

[14] Salari, S. R., & Rostami, H. (2016). Pgu-Face: A dataset of partially covered facial images. Data in Brief, 9, 288–291.

[15] Song, L., Gong, D., Li, Z., Liu, C., & Liu, W. (2019). Occlusion Robust Face Recognition Based on Mask Learning With Pairwise Differential Siamese Network. 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 773-782.*

[16] Wang, Z., Wang, G., Huang, B., Xiong, Z., Hong, Q., Wu, H., Yi, P., Jiang, K., Wang, N., Pei, Y., Chen, H., Miao, Y., Huang, Z., & Liang, J. (2020). Masked Face Recognition Dataset and Application. ArXiv, abs/2003.09093.

[17] Nair, A., & Potgantwar, A. (2018). Masked Face Detection using the Viola-Jones Algorithm: A Progressive Approach for less Time Consumption. Int. J. Recent Contributions Eng. Sci. IT, 6, 4-14.

[18] Ejaz, M.S., Islam, M.N., Sifatullah, M., & Sarker, A. (2019). Implementation of Principal Component Analysis on Masked and Non-masked Face Recognition. 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 1-5.

[19] Hariri, W. (2020). Efficient Masked Face Recognition Method during the COVID-19 Pandemic.

[20] Dey, S.K., Howlader, A., & Deb, C. (2021). MobileNet Mask: A Multi-phase Face Mask Detection Model to Prevent Person-To-Person Transmission of SARS-CoV-2.

[21] Loey, M., Manogaran, G., Taha, M., & Khalifa, N. (2021). A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. Measurement, 167, 108288

[22] Chen, Ququ & Sang, Lei. (2018). Face-Mask Recognition for Fraud Prevention Using Gaussian Mixture Model. Journal of Visual Communication and Image Representation. 55.

[23] Kim, Min & Koo, Ja & Cho, Se & Baek, Na & Park, Kang. (2018). Convolutional Neural Network-based Periocular Recognition in Surveillance Environments. IEEE Access. PP. 1-1.

[24] Zhao, Z., & Kumar, A. (2018). Improving Periocular Recognition by Explicit Attention to Critical Regions in Deep Neural Network. IEEE Transactions on Information Forensics and Security, 13(12), 2937–2952.

[25] Niu, G., & Chen, Q. (2018). Learning an video frame-based face detection system for security fields. Journal of Visual Communication and Image Representation, 55, 457–463.

[26] Elmahmudi, A., & Ugail, H. (2019). Deep face recognition using imperfect facial data. Future Generation Computer Systems, 99: 213-225

[27] Hosni Mahmoud, H. A., & Mengash, H. A. (2020). A novel technique for automated concealed face detection in surveillance videos. Personal and Ubiquitous Computing.

[28] Roy, Biparnak & Nandy, Subhadip & Ghosh, Debojit & Dutta, Debarghya & Biswas, Pritam & Das, Tamodip. (2020). MOXA: A Deep Learning Based Unmanned Approach for Real-Time Monitoring of People Wearing Medical Masks. Transactions of the Indian National Academy of Engineering.iong,

[29] Leslie & Lee, Yunli & Teoh, Andrew. (2019). Periocular Recognition in the Wild: Implementation of RGB-OCLBCP Dual-Stream CNN. Applied Sciences. 9. 2709.